Minimum-Distance Problems in Protocol Design

E. C. Posner
Office of Telecommunications and Data Acquisition

Z. Reichstein[1]
California Institute of Technology, Student

This article considers codes for use in personal computer file transfer as control characters, when only upper-case ASCII can be used to avoid dependence on unique machine features and promote portability. If ten control functions are needed, a number used in at least one protocol, we seek a subset of ten upper-case ASCII characters with good distance properties. The control functions form themselves naturally into three groups, one of two functions (ACK and NAK) and two of four. We wish to make ACK and NAK as antipodal as possible (distance 6), make the distances within each of the other groups as large as possible (4), and otherwise have as few 2's in the distance table as possible, recognizing that only even distances can occur. We find the minimum and an assignment that attains the minimum. The code is essentially unique. We also solve the analogous problem for two groups of three control functions and one group of four.

I. Introduction

Here we solve a simple coding problem in computer protocol design. People with personal computers want to exchange binary files, but everyone owns a different machine. The problem is that different machines interpret binary seventuples differently. So when setting up communication, it may be impossible to even pass the initial protocol characters to the protocol-handling software. But there is a way out of this. All personal computers agree on what to do with upper-case ASCII. So in this article we restrict protocol control functions to upper-case ASCII. This permits communication to be established between the two computers, after which the files can then be transferred as straight binary files.

Thus, we are in the following situation. We want to choose upper-case ASCII subsets with good distance properties. Before attempting this, we will note that ASCII here is actually 7 bits with even parity adjoined. However, two of the 8 bits are always equal in the upper-case alphabet. This means that we really have a 6-bit code to choose (see Table 1).

Let us formulate this as a precise coding problem. A typical protocol has 10 "block control functions." Here we choose to divide them into three groups, as in Table 2, a "2, 4, 4, Table."

The rationale is that functions within groups need extra protection from each other. This is because within groups, the control functions are more likely to be confusable because the protocol states typically result in outputs that can be or tend to be in only one group. Furthermore, ACK and NAK need to be as unconfusable as possible, to prevent false file transfers.

[1] Currently a graduate student at Harvard University

Other groupings are possible, for example a "3, 3, 4 Table" with NAK, ACK, QRY in Group 1, WT, ABH, ABW in Group 2, and EOD, EOT, SOH, SOD in Group 3. We shall consider this case as well.

Let us now formulate this as a coding problem. We are to choose a subset of 10 of the 26 upper-case ASCII odd-parity six-tuples (odd parity, because the invariable 10 has been removed), with the following properties:

a) The distance of ACK and NAK is 6.

b) The minimum distance within Groups 1 and 2 should be as large as possible.

c) There should be as few distances of 2 as possible.

This is for the 2, 4, 4 case. For the 3, 3, 4 case, we want the distances within groups to all be at least 4 (this makes them all 4, it turns out), with as few distances of 2 as possible. We will call a set of three code words of mutual distance 4 an *equilateral triangle* and of four code words of mutual distance 4 a *regular tetrahedron*. We note here that for both problems, it turns out that the restriction to the 26 upper-case ASCII out of the 32 odd-parity six-tuples did not hurt the distance table any, as it turned out in these two cases. Thus we will talk about the alphabet restriction no longer. Probabilistic arguments can be given that make this not too surprising.

II. Intermediate Results

Here we will for convenience revert to even-parity six-tuples instead of odd parity. Call this set E. We shall use lower-case Greek for the elements of E. Note that the Hamming distance d between any two elements of E is even. This section presents five propositions needed in deriving the optimal codes.

We want to solve the following two problems:

Problem 1: Find $\alpha_1, \alpha_2; \beta_1, \beta_2, \beta_3, \beta_4; \gamma_1, \gamma_2, \gamma_3, \gamma_4 \in \mathrm{E}$ such that

$$\left.\begin{array}{lll} d(\alpha_1, \alpha_2) = 6 & & \\ d(\beta_i, \beta_j) \geqslant 4, & i, j = 1, 2, 3, 4, & i \neq j \\ d(\gamma_i, \gamma_j) \geqslant 4, & i, j = 1, 2, 3, 4, & i \neq j \end{array}\right\} \quad (1)$$

and the number of 2's in the distance table of $\alpha_1, \alpha_2; \beta_1, \beta_2, \beta_3, \beta_4; \gamma_1, \gamma_2, \gamma_3, \gamma_4$ is minimal.

Problem 2: Find $\alpha_1, \alpha_2, \alpha_3; \beta_1, \beta_2, \beta_3; \gamma_1, \gamma_2, \gamma_3, \gamma_4 \in \mathrm{E}$ such that

$$\left.\begin{array}{lll} d(\alpha_i, \alpha_j) \geqslant 4, & i, j = 1, 2, 3, & i \neq j \\ d(\beta_i, \beta_j) \geqslant 4, & i, j = 1, 2, 3, & i \neq j \\ d(\gamma_i, \gamma_j) \geqslant 4, & i, j = 1, 2, 3, 4, & i \neq j \end{array}\right\} \quad (2)$$

and the number of 2's in the distance table of $\alpha_1, \alpha_2, \alpha_3; \beta_1, \beta_2, \beta_3; \gamma_1, \gamma_2, \gamma_3, \gamma_4$ is minimal.

Five propositions will be useful in our search.

Proposition 1: Let $\alpha_1, \alpha_2, \alpha_3 \in \mathrm{E}$ and $d(\alpha_1, \alpha_2), d(\alpha_2, \alpha_3), d(\alpha_1, \alpha_3) \geqslant 4$. Then $d(\alpha_1, \alpha_2) = d(\alpha_2, \alpha_3) = d(\alpha_1, \alpha_3) = 4$.

Proof: Assume the contrary. Say $d(\alpha_1, \alpha_2) > 4$, i.e., = 6. Then $\alpha_2^i = \overline{\alpha_1^i}$ where $\overline{x}$ means $\overline{1 - x}$ for $x = 0, 1$. Thus, for any $a \in \{0, 1\}, |a - \alpha_2^i| = \overline{|a - \alpha_1^i|}$ (i.e., if one of them is 0, then the other is one and vice versa) and, hence, $|a - \alpha_2^i| + |a - \alpha_1^i| = 1$. Then

$$8 \leqslant d(\alpha_1, \alpha_3) + d(\alpha_2, \alpha_3) = \sum_{i=1}^{6} (|\alpha_3^i - \alpha_2^i| + |\alpha_3^i - \alpha_1^i|)$$

$$= \sum_{i=1}^{6} 1 = 6$$

This contradiction proves the proposition.

Proposition 1 shows that we can replace "$\geqslant 4$" by "$= 4$" everywhere in Problems (1) and (2).

Proposition 2: Let $A = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$ be a regular tetrahedron. Then each column of the matrix

$$\begin{array}{cccc} \gamma_1^{(1)} & \gamma_1^{(2)} & \cdots & \gamma_1^{(6)} \\ \gamma_2^{(1)} & & \cdots & \gamma_2^{(6)} \\ \vdots & & & \\ \gamma_4^{(1)} & & \cdots & \gamma_4^{(6)} \end{array}$$

has exactly two 0's and two 1's in it, and is unique up to permuting and conjugating (complementing modulo 2) columns.

Proof: We can assume without loss of generality that $\gamma_1 = (0, 0, 0, 0, 0, 0)$; otherwise just conjugate the columns where

$\gamma_i^{(1)} = 1$, i.e., replace 0 by 1 and 1 by 0 in those columns. Then γ_2, γ_3, and γ_4 have exactly two 0's and four 1's. By permuting the columns we can make $\gamma_2 = (1, 1, 1, 1, 0, 0)$. Now γ_3 and γ_4 must have their last two positions be 11. Otherwise their distance from γ_2 would be at most 2. By permuting the first four columns we can make $\gamma_3 = (1, 1, 0, 0, 1, 1)$. Since $d(\gamma_3, \gamma_4) = 4$, γ_4 must be equal to $(0, 0, 1, 1, 1, 1)$.

Thus any regular tetrahedron can be obtained from

$$\begin{aligned} (0,0,0,0,0,0) &= \gamma_1 \\ (1,1,1,1,0,0) &= \gamma_2 \\ (1,1,0,0,1,1) &= \gamma_3 \\ (0,0,1,1,1,1) &= \gamma_4 \end{aligned}$$

by permuting and conjugating columns.

Since these operations preserve the property we are interested in (exactly 2 zeros in every column), it is sufficient to check it just for the regular tetrahedron above. It, indeed, has exactly two 0's in each column. Hence, any regular tetrahedron also has this property. (Note that we can only conjugate an even number of columns if the tetrahedron is to remain a subset of E.)

Proposition 2 also follows from a more general result on constant-distance codes, but we shall not do it this way.

Proposition 3: Let $B = \{\beta_1, \beta_2, \beta_3\}$ be an equilateral triangle. Then the matrix

$$\begin{matrix} \beta_1^{(1)} & \beta_1^{(2)} & \cdots & \beta_1^{(6)} \\ \beta_2^{(1)} & & \cdots & \beta_2^{(6)} \\ \beta_3^{(1)} & & \cdots & \beta_3^{(6)} \end{matrix}$$

has either one 0 and two 1's or one 1 and two 0's in each column, and is unique up to permuting and conjugating columns.

Proof: We apply the same argument as in Proposition 2. The property we are interested in is again invariant under conjugations and permutations of columns. This operation allows us to transform any equilateral triangle into

$$\begin{aligned} (0,0,0,0,0,0) &= \gamma_1 \\ (1,1,1,1,0,0) &= \gamma_2 \\ (1,1,0,0,1,1) &= \gamma_3 \end{aligned}$$

as we showed in the proof of Proposition 2. It is easy to see that γ_1, γ_2, and γ_3 have the desired property.

Proposition 4: Let $C = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$ be a regular tetrahedron of elements in E, and let γ also be in E. Then

$$\sum_{i=1}^{4} d(\gamma, \gamma_i) = 12$$

Proof: Proposition 2 implies that for $a \in \{0, 1\}$, $k = 1, 2, \ldots, 6$, we have

$$\sum_{i=1}^{4} |a - \gamma_i^{(k)}| = 2$$

for each column k. Thus

$$\sum_{i=1}^{4} |\gamma^{(k)} - \gamma_i^{(k)}| = 2, \qquad 1 \leqslant k \leqslant 6$$

Adding these six equalities together, we get, as stated,

$$\begin{aligned} 12 &= \sum_{k=1}^{6} \left(\sum_{i=1}^{4} |\gamma^{(k)} - \gamma_i^{(k)}| \right) \\ &= \sum_{i=1}^{4} \left(\sum_{k=1}^{6} |\gamma^{(k)} - \gamma_i^{(k)}| \right) \\ &= \sum_{i=1}^{4} d(\gamma, \gamma_i) \end{aligned}$$

Proposition 5. Let $A = \{\alpha_1, \alpha_2, \alpha_3\}$, $B = \{\beta_1, \beta_2, \beta_3\}$ be two equilateral triangles of elements of E. Then

$$\sum_{i,j=1,2,3} d(\alpha_i, \beta_j) \leqslant 30$$

If the sum is 30, then each

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| = 5$$

for $1 \leqslant k \leqslant 6$ (used for exhaustive search).

Proof: First we show that

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| \leqslant 5$$

for each $k = 1, 2, \ldots, 6$. The sum

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}|$$

does not change when we simultaneously conjugate all the kth components

$$\alpha_1^{(k)}, \alpha_2^{(k)}, \alpha_3^{(k)}, \beta_1^{(k)}, \beta_2^{(k)}, \beta_3^{(k)}$$

and permute

$$\{\alpha_1^{(k)}, \alpha_2^{(k)}, \alpha_3^{(k)}\}$$

or

$$\{\beta_1^{(k)}, \beta_2^{(k)}, \beta_3^{(k)}\}$$

for each $k = 1, 2, \ldots, 6$.

Thus by Proposition 3 we can assume that $\alpha_1^{(k)}, \alpha_2^{(k)} = 0$ and $\alpha_3^{(k)} = 1$. Then (again by Proposition 3) one of the $\beta_i^{(k)}$ ($i = 1, 2, 3$) is 0 and one of them is 1. Number the 0 first (i.e., $\beta_1^{(k)} = 0$) and the 1 last ($\beta_3^{(k)} = 0$). Now we only have two choices: $\beta_2^{(k)} = 0$ and $\beta_2^{(k)} = 1$. In the first case

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| = 4$$

and in the second case

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| = 5$$

Thus

$$\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| \leqslant 5$$

for each k from 1 to 6. Adding these six inequalities together, we get, as stated,

$$30 \geqslant \sum_{k=1}^{6} \left(\sum_{i,j=1,2,3} |\alpha_i^{(k)} - \beta_j^{(k)}| \right)$$

$$= \sum_{i,j=1,2,3} \left(\sum_{k=1}^{6} |\alpha_i^{(k)} - \beta_j^{(k)}| \right)$$

$$= \sum d(\alpha_i, \beta_j)$$

The second part of the proposition also follows.

III. The Optimal Codes

Figure 1 gives the optimal odd-parity 8-bit ASCII upper-case alphabetic code for the 3, 3, 4 case (2) and Fig. 2 an optimal code for the 2, 4, 4 case (1). Why are they optimal, even when we drop the alphabetic restriction and would be willing to allow *any* even-parity six tuples?

Suppose we have a distance table satisfying (2), with equalities instead of inequalities by Proposition 1. Break it up into blocks as shown in Fig. 3. By Proposition 4 each row sum within Blocks II and III must be 12. Since every entry is 2, 4, or 6, this means that each row in Blocks II and III contains at least two 2's. Thus the total number of 2's in Block II is at least 6 and the total number of 2's in Block III is at least 6.

By Proposition 5 the sum of the nine entries in Block I is at most 30. This implies that Block I contains at least three 2's ($7 \times 4 + 2 \times 2 = 32 > 30$). Blocks I′, II′, and III′ are just transposes of Blocks I, II, and III, respectively, and, hence, always have the same number of 2's as Blocks I, II, and III. We see that Fig. 1 has the minimal possible number of 2's in each block. This shows that Fig. 1 is a solution for Problem 2. There are thirty 2's in this table.

The same argument (see Fig. 4) shows that any distance table satisfying (1) must have at least four 2's in Block I, four 2's in Block II, and eight 2's in Block III ($7 \times 2 + 9 \times 4 = 50 > 48$). This proves that Fig. 2 has the minimal possible number of 2's in each block and is a solution for Problem 1. There are thirty-two 2's in this table.

If 20 control functions are desired instead of 10, we can use *pairs* of upper-case letters. We have found a constant distance 6 code that does this for 20 functions. This will be reported elsewhere in a more general context.

IV. Uniqueness Results

The code of Fig. 2 with its three groups has a unique automorphism crossing group boundaries. In this, O, H exchange with K, L (or Q, V with U, R), with O corresponding to L and H corresponding to K. Similarly, in Fig. 1, exchange X and Z between groups. This forces V to interchange with O and L with K. Also, in each problem, the two groups of equal size can be swapped.

Are our solutions unique up to the obvious operations? The answer is *yes* for the 2, 4, 4 problem. This is easier to prove than to write up, and we will merely assert it.

The answer is *no* for the 3, 3, 4 problem. In fact, a different distance table can even be achieved, although it can be shown that the two triangles are unique up to obvious transformations. Merely interchange U and A in Fig. 1, and observe that Block II (upper right) becomes the inequivalent (because of the column of all 4's) block

$$\begin{array}{cccc} 2 & 4 & 2 & 4 \\ 4 & 2 & 2 & 4 \\ 2 & 2 & 4 & 4 \end{array}$$

This is also inequivalent to Block III, so it is a really different distance table. We shall say no more about these uniqueness problems.

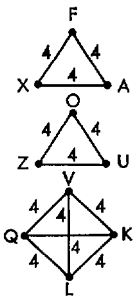
Table 1. Upper case ASCII with even parity

A	0100	0001
B	0100	0010
C	1100	0011
D	0100	0100
E	1100	0101
F	1100	0110
G	0100	0111
H	0100	1000
I	1100	1001
J	1100	1010
K	0100	1011
L	1100	1100
M	0100	1101
N	0100	1110
O	1100	1111
P	0101	0000
Q	1101	0001
R	1101	0010
S	0101	0011
T	1101	0100
U	0101	0101
V	0101	0110
W	1101	0111
X	1101	1000
Y	0101	1001
Z	0101	1010

Table 2. Three groups of block control functions in the 2,4,4 case

Function	Meaning
	Group 0
ACK	Acknowledgement
NAK	Negative acknowledgment
	Group 1
EOD	End of data
EOT	End of text
SOH	Start of header
SOD	Start of data
	Group 2
WT	Wait
QRY	Query
ABH	Abort and hang up
ABW	Abort and wait

	FUNCTION	CHARACTER	8-BIT ASCII
GROUP 0	ACK	F	01000110
	NAK	X	01011000
	WT	A	11000001
GROUP 1	QRY	O	01001111
	ABH	Z	11011010
	ABW	U	11010101
GROUP 2	EOD	V	11010110
	EOT	Q	01010001
	SOH	L	01001100
	SOD	K	11001011



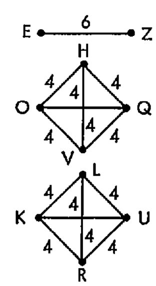
		DISTANCE MATRIX GROUP 0	DISTANCE MATRIX GROUP 1	DISTANCE MATRIX GROUP 2
GROUP 0	ACK (F)	0 4 4	2 4 4	2 4 2 4
	NAK (X)	4 0 4	4 2 4	4 2 2 4
	WT (A)	4 4 0	4 4 2	4 2 4 2
GROUP 1	QRY (O)	2 4 4	0 4 4	4 4 2 2
	ABH (Z)	4 2 4	4 0 4	2 4 4 2
	ABW (U)	4 4 2	4 4 0	2 2 4 4
GROUP 2	EOD (V)	2 4 4	4 2 2	0 4 4 4
	EOT (Q)	4 2 2	4 4 2	4 0 4 4
	SOH (L)	2 2 4	2 4 4	4 4 0 4
	SOD (K)	4 4 2	2 2 4	4 4 4 0

3 TWO's PER ROW

3 TWO's

Fig. 1. Optimal code for the 3,3,4 case

	FUNCTION	CHARACTER	8-BIT ASCII
GROUP 0	ACK	E	01000101
	NAK	Z	11011010
GROUP 1	EOD	O	01001111
	EOT	H	11001000
	SOH	Q	01010001
	SOD	V	11010110
GROUP 2	WT	K	11001011
	QRY	L	01001100
	ABH	U	11010101
	ABW	R	01010010



		DISTANCE MATRIX GROUP 0	DISTANCE MATRIX GROUP 1	DISTANCE MATRIX GROUP 2
GROUP 0	ACK (E)	0 6	2 4 2 4	4 2 2 4
	NAK (Z)	6 0	4 2 4 2	2 4 4 2
GROUP 1	EOD (O)	2 4	0 4 4 4	2 2 4 4
	EOT (H)	4 2	4 0 4 4	2 2 4 4
	SOH (Q)	2 4	4 4 0 4	4 4 2 2
	SOD (V)	4 2	4 4 4 0	4 4 2 2
GROUP 2	WT (K)	4 2	2 2 4 4	0 4 4 4
	QRY (L)	2 4	2 2 4 4	4 0 4 4
	ABH (U)	2 4	4 4 2 2	4 4 0 4
	ABW (R)	4 2	4 4 2 2	4 4 4 0

4 TWO's PER ROW

3 TWO's PER ROW

4 TWO's   3 TWO's

Fig. 2. Optimal code for the 2,4,4 case

	$\alpha_1\ \alpha_2\ \alpha_3$	$\beta_1\ \beta_2\ \beta_3$	$\gamma_1\ \gamma_2\ \gamma_3\ \gamma_4$
α_1 α_2 α_3	**4**	BLOCK I	BLOCK II
β_1 β_2 β_3	BLOCK I'	**4**	BLOCK III
γ_1 γ_2 γ_3 γ_4	BLOCK II'	BLOCK III'	**4**

Fig. 3. Block structure for 3,3,4

	$\alpha_1\ \alpha_2$	$\beta_1\ \beta_2\ \beta_3\ \beta_4$	$\gamma_1\ \gamma_2\ \gamma_3\ \gamma_4$
α_1 α_2	**6**	BLOCK I	BLOCK II
β_1 β_2 β_3 β_4	BLOCK I'	**4**	BLOCK III
γ_1 γ_2 γ_3 γ_4	BLOCK II'	BLOCK III'	**4**

Fig. 4. Block structure for 2,4,4